

2026 “钉耙编程” 中国大学生算法设计暑期联赛

第一场

2026 年 7 月 21 日



题目列表

A	小丑牌
B	画树
C	边界主义
D	搭积木
E	摩卡数
F	开关灯
G	简单几何题
H	数字子序列
I	数组
J	游戏
K	森林
L	向量
M	彩色匹配

本试题册共有 **13** 题，**28** 页。

比赛正式开始前不得打开

此页故意留空



Problem A. 小丑牌

输入：标准输入
输出：标准输出
时间限制：2 秒
空间限制：512 MiB

Kasumi 正在玩“小丑牌”游戏。

游戏中有两个计分参数 x, y 。同时，Kasumi 有两类技能卡可以选择。每一张技能卡上都有参数，第 i 张第一类技能卡上的参数为 a_i ，第 i 张第二类技能卡上的参数为 b_i 。

- 第一类技能卡共有 n 张，若选择第 i 张第一类技能卡，则 x 变为 $x + a_i$ ；
- 第二类技能卡共有 m 张，若选择第 i 张第二类技能卡，则 y 变为 $y + b_i$ 。

Kasumi 可以选择至多一张第一类技能卡和至多一张第二类技能卡。若 Kasumi 没有选择某个类型的技能卡，则对应的计分参数保持不变。

选择完技能卡后，Kasumi 在游戏中的得分为 $x \times y$ 。Kasumi 想知道通过选择技能卡，在游戏中能获得的最大得分。

输入格式

第一行输入一个正整数 T ($1 \leq T \leq 10^5$)，表示数据组数。接下来按如下格式输入 T 组数据：

第一行输入四个整数 x, y, n, m ($|x|, |y| \leq 10^9, 0 \leq n, m \leq 10^5$)。

第二行输入 n 个整数 $a_1, a_2, \dots, a_n$ ($|a_i| \leq 10^9$)。若 $n = 0$ ，则输入的第二行为空行。

第三行输入 m 个整数 $b_1, b_2, \dots, b_m$ ($|b_i| \leq 10^9$)。若 $m = 0$ ，则输入的第三行为空行。

保证输入数据中 $\sum n, \sum m$ 均不超过 10^6 。

输出格式

对于每组数据，输出一行一个整数，表示通过选择技能卡可能达到的最高得分。

样例

标准输入	标准输出
3 1 2 3 4 -1 -2 -3 4 5 6 7 10 -20 5 4 1 2 3 4 -11 -5 -6 -7 -8 1000000000 -1000000000 3 3 -100000 -10000000 -1000000000 100000 10000000 1000000000	9 28 0



提示

本题输入输出量较大，建议使用较快速的输入输出方式（如关闭流同步的 `cin / cout`）。



Problem B. 画树

输入：标准输入
输出：标准输出
时间限制：5 秒
空间限制：512 MiB

众所周知，Arisa 是盆栽大王。今天，Arisa 希望在画布上画出利根川的结构。

利根川是一棵有 n 个节点的有根二叉树，根的编号固定为 1，保证对于所有节点 i 和 i 的父结点 f_i 有 $f_i < i$ 。设节点 i 的左儿子和右儿子分别为 L_i 和 R_i 。

Arisa 需要给每个节点分配一个**整数坐标** $U_i = (x_i, y_i)$, ($|x_i|, |y_i| \leq 10^8$)，满足：

- 对于任意两个节点，分配的坐标不能相同；
- 对于所有二叉树中的边 (u, v) ($u < v$)，满足 $y_u < y_v$ ；
- 对于二叉树中任意两条边 $(a, b), (c, d)$ ，线段 $U_a U_b$ 和 $U_c U_d$ 不允许相交或重合，但允许共用端点；
- 对于二叉树中任意节点 i ， i 的左儿子（若存在）的 x 坐标必须不大于 i 的 x 坐标；且 i 的右儿子（若存在）的 x 坐标必须不小于 i 的 x 坐标。即 $x_{L_i} \leq x_i \leq x_{R_i}$ 。

然而，画布的大小是有限的。Arisa 使用的画布大小不能超过 **66666666**。即：

$$\left(\max_{i=1}^n x_i - \min_{i=1}^n x_i \right) \times \left(\max_{i=1}^n y_i - \min_{i=1}^n y_i \right) \leq 66666666$$

你需要帮助 Arisa 找到一个坐标分配方案，使得上述所有限制都被满足。

可以证明，在题目给定的数据范围内，对于任意二叉树都一定存在至少一种满足上述所有条件的分配方案。

输入格式

本题单个测试点内有多组测试数据。

第一行输入一个正整数 T ($1 \leq T \leq 50$)，表示数据组数。

接下来按如下格式输入 T 组数据：

第一行输入一个正整数 n ($1 \leq n \leq 10^5$)，表示二叉树的节点个数。

接下来输入 n 行，其中第 i 行输入两个整数 L_i 和 R_i ($0 \leq L_i, R_i \leq n$)，表示节点 i 的左儿子和右儿子。若节点 i 没有左儿子，则 $L_i = 0$ ；若节点 i 没有右儿子，则 $R_i = 0$ 。

保证输入的树是一棵以 1 为根的二叉树。

输出格式

对于每组数据：

输出 n 行，其中第 i 行输出两个用空格分隔的整数 x_i, y_i ，表示分配给节点 i 的坐标。

你需要保证 $|x_i|, |y_i| \leq 10^8$ 。



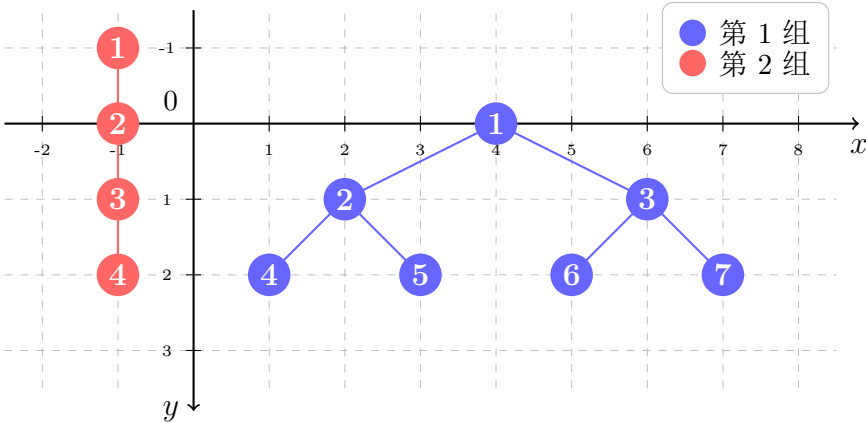
本题使用 Special Judge 进行评测。如有多种满足所有条件的坐标分配方案，你可以输出其中的任意一种。

样例

标准输入	标准输出
2	4 0
7	2 1
2 3	6 1
4 5	1 2
6 7	3 2
0 0	5 2
0 0	7 2
0 0	-1 -1
0 0	-1 0
4	-1 1
2 0	-1 2
0 3	
4 0	
0 0	

提示

下图绘制了样例数据中的两组坐标分配方案。其中第 1 组数据占用的面积为 $(7 - 1) \times (2 - 0) = 12$ ，第 2 组数据占用的面积为 $(-1 - (-1)) \times (2 - (-1)) = 0$ 。



样例输出仅表示一种可能的答案，并不表示该样例输出恰好对应标准程序的输出。
本题输入输出量较大，建议使用较快速的输入输出方式（如关闭流同步的 `cin / cout`）。



Problem C. 边界主义

输入：标准输入
输出：标准输出
时间限制：15 秒
空间限制：128 MiB

请注意本题不同寻常的空间限制。

Rimi 正在制作 PopiV。为此，她开始研究一种叫作**边界主义**的电子抽象画派。

在边界主义中，画家不会直接描绘具体物体，而是从画布的左右边缘拖出色块，让它们在画布上互相覆盖。Rimi 准备在一张大小为 $n \times n$ 的电子画布上作画。画布由 n^2 个像素组成，其中第 x 列、第 y 行的像素记为 (x, y) 。

每个像素都有一个整数亮度 $V(x, y)$ 。最开始，所有像素的亮度均为 0。

Rimi 会进行恰好 n 次上色操作。每次上色操作会给出五个整数 x_1, x_2, y_1, y_2, v ，这表示她选中矩形区域 $[x_1, x_2] \times [y_1, y_2]$ 中的所有像素，并将它们的亮度提高到至少 v 。

形式化地，对于所有满足 $x_1 \leq x \leq x_2, y_1 \leq y \leq y_2$ 的像素 (x, y) ，执行 $V(x, y) \leftarrow \max(V(x, y), v)$ 。

由于边界主义的作画规则，每个色块必须从画布的左边界或右边界延伸出来。因此，对于每一次上色操作，都保证 $x_1 = 1$ 或者 $x_2 = n$ 。

完成所有上色操作后，Rimi 想要检查画面中若干个矩形区域的整体亮度。

她会提出 m 次询问。每次询问给出四个整数 x_1, x_2, y_1, y_2 ，你需要回答矩形区域中所有像素最终亮度之和，即 $\sum_{x=x_1}^{x_2} \sum_{y=y_1}^{y_2} V(x, y)$ 。

输入格式

第一行一个整数 T ($1 \leq T \leq 50$)，表示数据的组数。对于每组数据：

第一行两个整数 n, m ($1 \leq n, m \leq 2 \times 10^5$)；

接下来 n 行每行五个整数 x_1, x_2, y_1, y_2, v ，依次表示每次上色操作；

接下来 m 行每行四个整数 x_1, x_2, y_1, y_2 ，依次表示每次查询操作；

每个修改或查询操作均满足 $1 \leq x_1 \leq x_2 \leq n, 1 \leq y_1 \leq y_2 \leq n$ 。

每个修改操作均满足 $1 \leq v \leq n$ ，且保证 $x_1 = 1$ 或者 $x_2 = n$ 。

对于所有数据，保证 n 的总和和 m 的总和均不超过 10^6 。

输出格式

对于每组数据，输出 m 行，依次表示每次查询操作的答案。



样例

标准输入	标准输出
1	111
10 10	542
1 3 4 5 9	20
8 10 4 8 10	41
1 10 2 4 7	187
8 10 2 8 6	116
10 10 4 7 2	17
1 4 1 9 2	90
1 3 1 4 9	400
1 10 8 9 7	42
3 10 5 6 3	
6 10 1 6 2	
8 10 3 6	
1 10 2 9	
9 9 4 5	
2 3 4 6	
1 10 7 10	
2 9 4 5	
9 9 3 4	
1 2 3 10	
1 8 2 9	
4 6 8 10	

提示

本题输入输出量较大，建议使用较快速的输入输出方式（如关闭流同步的 `cin / cout`）。



Problem D. 搭积木

输入：标准输入
输出：标准输出
时间限制：3 秒
空间限制：512 MiB

小 W 正在用积木搭建一个巨大的结构。

有 n 块积木，编号为 1 到 n 。第 i 块积木有两个整数属性 a_i 和 b_i 。

一开始，小 W 打算一块一块地搭建。对每块积木 i ，给定一个积木 f_i ，表示积木 i 必须直接放在积木 f_i 的上方。如果 $f_i = 0$ ，则积木 i 位于整个结构的最底部。题目保证这些要求是自洽的，并且最终会形成一个包含所有积木的完整结构。

后来，小 W 觉得逐块搭建太无聊了。他想到一种新方法：可以先把若干块积木组装成一个结构，再一次性把整个已组装结构放到另一个结构上。具体地，对于两个不交的连通集合 A 和 B ，若 A 中存在唯一的点 x 使得 $f_x \in B$ ，那么 A 就可以被放在 B 的上面。

也就是说，在满足所有原始直接上下关系的前提下，小 W 可以自由选择执行放置操作的顺序，也可以提前组装某些局部结构。

一次放置操作中，假设一个已经组装好的结构被放到另一个结构的上方。令：

- A 为上方结构内所有积木的 a_i 之和；
- B 为下方承托结构内所有积木的 b_i 之和。

这次放置操作的代价定义为 $A \times B$ 。

操作完成后，两个结构会合并为一个更大的结构。

注意，下方承托结构指的是当前已经组装在一起的整个结构，而不只是单块积木或它的祖先。

小 W 想知道，在所有合法的搭建顺序中，最小总代价是多少。

输入格式

本题单个测试点内有多组测试数据。

第一行输入一个正整数 T ($1 \leq T \leq 20$)，表示测试数据组数。

接下来按如下格式输入 T 组数据：

第一行输入一个整数 n ($1 \leq n \leq 2 \times 10^5$)，表示积木数量。保证单个测试点内所有测试数据的 n 之和不超过 10^6 。

第二行输入 n 个整数 $a_1, a_2, \dots, a_n$ ($1 \leq a_i \leq 10^3$)。

第三行输入 n 个整数 $b_1, b_2, \dots, b_n$ ($1 \leq b_i \leq 10^3$)。

第四行输入 n 个整数 $f_1, f_2, \dots, f_n$ 。其中 $f_1 = 0$ ，对于所有 $i > 1$ ，保证 $1 \leq f_i < i$ 。

输出格式

对于每组数据，输出一行一个整数，表示搭建完整结构所需的最小总代价。



样例

标准输入	标准输出
1 3 3 10 1 5 1 10 0 1 1	56

提示

样例中，积木 2 和 3 都必须直接放在积木 1 上方。
如果先把积木 2 放到积木 1 上，代价为 $10 \times 5 = 50$ 。
此时下方结构包含积木 $\{1, 2\}$ ，其 b 之和为 $5 + 1 = 6$ 。再放置积木 3，代价为 $1 \times 6 = 6$ 。
总代价为 $50 + 6 = 56$ ，可以证明这是最优的。
本题输入量较大，建议使用较快速的输入方式（如关闭流同步的 `cin`）。



Problem E. 摩卡数

输入：标准输入
输出：标准输出
时间限制：1 秒
空间限制：512 MiB

小摩卡是个天才，尤其在字符串理论方面有着异于常人的天赋。为了赞颂她的才华，人们常常将那些满足特定优美性质的字符串命名为“摩卡串”。

小摩卡上本科时，在数据结构与算法分析课程中学到了 KMP 自动机，并想出了如下构建一个字符串 S 的 KMP 自动机的算法。

在算法中：

- S 是长度为 n 的输入字符串，仅包含前 σ 个小写字母。
- π 是 S 的前缀函数。 $\pi[i]$ 的值是满足 $k < i$ 且 $S[1, k] = S[i - k + 1, i]$ 的最大 k 。若不存在这样的正整数 k ，则 $\pi[i] = 0$ 。
- $S[l, r]$ 表示仅保留 S 中第 l 个到第 r 个位置的字符所构成的子串。
- δ 是 KMP 自动机的转移表。 $\delta[i, j]$ 表示如果当前状态为 i 且输入字符为 j ，则新状态变为 $\delta[i, j]$ 。
- k 是一个计数器。

Algorithm 1 小摩卡的算法

Require: 字符串 S （下标从 1 开始），前缀数组 π ，长度 n ，字符集大小 σ

Ensure: 转移表 δ ，摩卡数 k

```
1:  $k \leftarrow 0$ 
2: for  $i \leftarrow 1$  to  $n - 1$  do
3:   for  $j \leftarrow 0$  to  $\sigma - 1$  do
4:      $c \leftarrow \text{char}(j)$  ▷ 字母表中第  $j + 1$  个小写字母
5:      $pos \leftarrow i$ 
6:     while  $pos > 0 \wedge c \neq S[pos + 1]$  do
7:        $pos \leftarrow \pi[pos]$ 
8:        $k \leftarrow k + 1$ 
9:     end while
10:    if  $c = S[pos + 1]$  then
11:       $\delta[i, j] \leftarrow pos + 1$ 
12:    else
13:       $\delta[i, j] \leftarrow pos$ 
14:    end if
15:  end for
16: end for
17: print  $k$ 
```

小摩卡定义，算法结束后 k 的值即为字符串 S 的摩卡数。请你构造一个字符串 S 并指定 S 的字符集大小 σ ，使得对字符串 S 运行该算法，所得到的摩卡数恰好为 k 。

你需要保证构造的字符串的长度不超过 10^5 。



输入格式

第一行输入一个正整数 T ($1 \leq T \leq 50$)，表示数据组数。接下来按如下格式输入 T 组数据：

输入一行一个正整数 k ($1 \leq k \leq 10^9$)，表示摩卡数的值。

输出格式

对于每组数据，输出两行：

- 第一行输出两个用空格分隔的正整数 n, σ ，表示字符串的长度和字符集的大小。
- 第二行输出一行一个字符串 S 。

你需要保证 $1 \leq n \leq 10^5, 1 \leq \sigma \leq 26$ ， S 的长度为 n ，且 S 仅包含前 σ 个小写英文字母。本题使用 Special Judge 测试，如有多个满足条件的答案，你可以输出任意一种。你不需要最小化字符串 S 的长度或字典序。

可以证明在题目限制内，一定存在至少一组满足条件的解。

样例

标准输入	标准输出
2 14 697	6 3 abcabc 21 26 cbababcbbabcbabcbabc

提示

请注意样例输出仅表示一种可能的合法答案，并不表示该样例输出恰好对应标准程序的输出。

本题输出量可能较大，建议使用较快速的输出方式（如关闭流同步的 `cout`）。



Problem F. 开关灯

输入：标准输入
输出：标准输出
时间限制：2 秒
空间限制：512 MiB

小 D 很喜欢随机性。

他面前有一排 n 盏灯，从左到右编号为 $1, 2, \dots, n$ ，第 i 盏灯的权值为 a_i 。一开始，所有灯都是关闭的。

小 D 随机选择一个 $1 \sim n$ 的排列，并按照这个排列依次打开所有灯。

每打开一盏灯后，所有亮着的灯会形成若干个极大连续段。若当前共有 c 个连续段，并且这次打开的是第 i 盏灯，那么小 D 会获得 $a_i \cdot c$ 分。

小 D 想知道，最终总得分的期望是多少？答案对 998 244 353 取模。

输入格式

本题有多组测试数据。第一行输入一个正整数 T ($1 \leq T \leq 25$)，表示测试数据组数。

接下来输入 T 组测试数据。每组测试数据包含两行：

第一行一个整数 n ($1 \leq n \leq 2 \times 10^5$)。

第二行 n 个整数 $a_1, a_2, \dots, a_n$ ($0 \leq a_i < 998\,244\,353$)。

保证单个测试点内所有测试数据的 n 之和不超过 2×10^6 。

输出格式

输出一个整数，表示小 D 最终总得分的期望值对 998 244 353 取模后的结果。

可以证明答案是一个分母不被 998 244 353 整除的有理数。若答案最简表示为 $\frac{x}{y}$ ，你需要输出 $x \times y^{-1} \bmod 998\,244\,353$ ，其中 y^{-1} 表示 y 在模 998 244 353 意义下的乘法逆元。

样例

标准输入	标准输出
2 1 10 3 1 2 3	10 665496242

提示

本题输入量较大，建议使用较快速的输入方式（如关闭流同步的 `cin`）。



此页故意留空

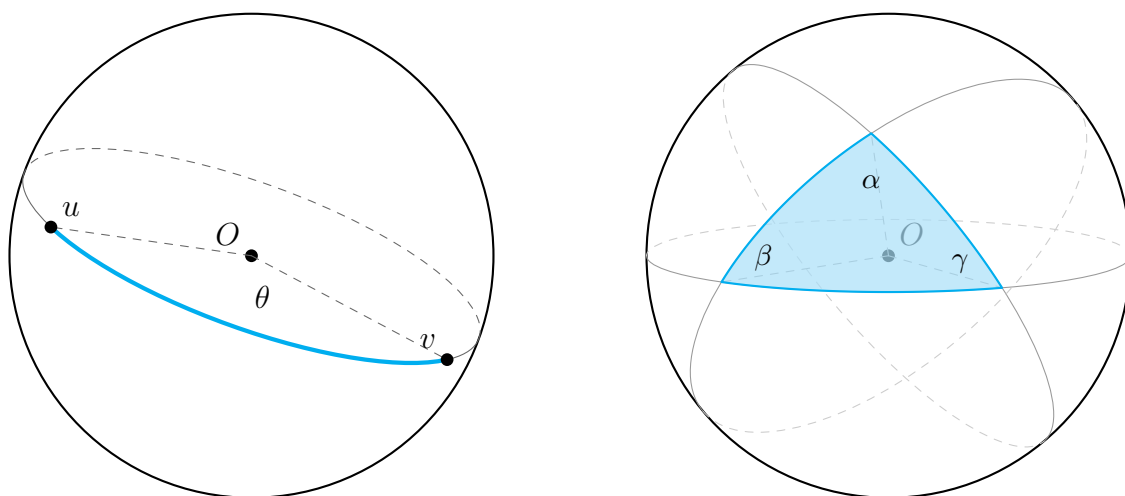


Problem G. 简单几何题

输入：标准输入
输出：标准输出
时间限制：2 秒
空间限制：512 MiB

小 D 生活在被无尽海洋覆盖的星球上。由于行星所处的恒星系统引力极其混沌，它的自转轴和公转轨道处于极度的不稳定状态。这导致太阳光直射的方向都是完全随机的。

行星的表面可以被视作三维欧几里得空间中的单位球面 $S^2 = \{(x, y, z) \in \mathbb{R}^3 : x^2 + y^2 + z^2 = 1\}$ 。连接球面上任意两点（不为对跖点）的大圆上的劣弧，被称为短大圆弧。



这颗星球上仅有一块古老且连续的大陆 P ， P 是一个球面简单多边形。它是由 n 条短大圆弧首尾相连构成的闭合区域，其边界不自交。同时，这块大陆整体被严格包含在某个开半球之内。

当太阳光在星球上的直射点为 s 时，星球上处于白天的区域被定义为 $H_s = \{x \in S^2 : s \cdot x > 0\}$ 。界定白天与黑夜的大圆 $s \cdot x = 0$ 称为晨昏线。

为了在变幻莫测的气候中生存，小 D 致力于建立精确的行星气象模型。小 D 的核心任务是：在给定大陆形状与任意太阳直射点 s 时，计算这块大陆有多少比例的面积恰好沐浴在阳光之中。即求出以下百分比：

$$\frac{\text{Area}(P \cap H_s)}{\text{Area}(P)} \cdot 100\%$$

球面面积： $\text{Area}(P)$ 指区域在单位球面上的表面积。在单位球上，一个边界大圆平面之间的二面角分别为 α, β, γ 的球面三角形，其球面面积等于 $\alpha + \beta + \gamma - \pi$ ，可结合上图理解。



输入格式

输入包含多组测试数据。第一行包含一个整数 T ($1 \leq T \leq 500$), 表示测试数据的组数。

对于每组测试数据:

第一行包含两个整数 n, q ($3 \leq n \leq 500, 1 \leq q \leq 500$), 分别表示大陆多边形的顶点数和查询的太阳方向数。

接下来的 n 行, 每行包含三个整数 x, y, z , 表示多边形的顶点。第 i 个顶点对应的实际球面坐标为 $\left(\frac{x}{\sqrt{x^2+y^2+z^2}}, \frac{y}{\sqrt{x^2+y^2+z^2}}, \frac{z}{\sqrt{x^2+y^2+z^2}}\right)$ 。顶点按相对于球体外侧的逆时针顺序给出。

接下来的 q 行, 每行包含三个整数 x, y, z , 表示一个查询的太阳直射点。对应的实际方向向量 s 为 $\left(\frac{x}{\sqrt{x^2+y^2+z^2}}, \frac{y}{\sqrt{x^2+y^2+z^2}}, \frac{z}{\sqrt{x^2+y^2+z^2}}\right)$ 。

额外保证

1. 所有测试数据的 $\sum nq \leq 5 \times 10^6$ 。
2. 输入的所有坐标均满足 $-10^5 \leq x, y, z \leq 10^5$ 且 $(x, y, z) \neq (0, 0, 0)$ 。
3. 令 $\gamma = 10^{-6}$ rad。对于多边形任意两个不同顶点 u, v , 它们之间的球面距离 $d(u, v)$ 满足 $\gamma \leq d(u, v) \leq \pi - \gamma$ 。
4. 多边形边界不自交; 相邻三点不共大圆; 多边形总面积保证 $\text{Area}(P) \geq 10^{-3}$ 。
5. 对于任意给定的直射点 s 和任意顶点 p_i , 保证点积 $s \cdot p_i \neq 0$, 即晨昏线不会穿过任何一个多边形顶点。
6. 存在向量 v 使得对任意 p_i 满足 $v \cdot p_i > 0$, 即多边形整体被包含在某个开半球之内。

输出格式

本题使用 Special Judge 进行评测。

对于每个查询, 输出一行包含一个实数 $r \in [0, 100]$, 表示白天部分占总面积的百分比。如果你的输出为 x , 标准答案为 y , 当且仅当满足 $\frac{|x-y|}{\max(1, |y|)} \leq 10^{-6}$ 时, 你的答案将被视为正确。

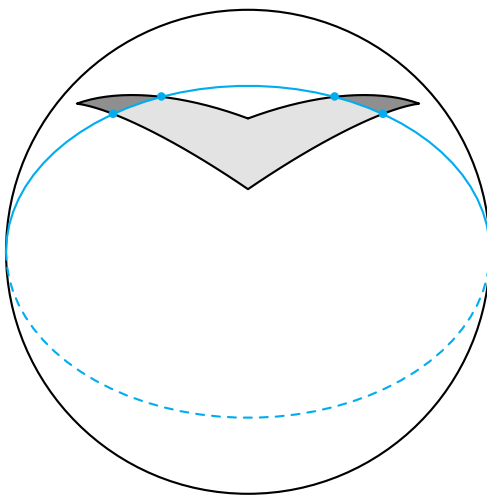
样例

标准输入	标准输出
1	16.37693771831994469373
4 2	83.62306228168005500789
10 0 10	
0 10 10	
-10 0 10	
0 5 10	
0 -10 3	
0 10 -3	



提示

下图为样例的俯视示意图，青色分界线为晨昏线（两次查询的方向恰好相反）。其中，深色区域为第一次查询对应的白天部分，而浅色区域为第二次查询对应的白天部分。



该多边形总面积的精确形式为 $4 \arctan \left(\frac{\sqrt{2}}{10+3\sqrt{10}} \right)$ 。

第一次询问所对应的区域面积的精确形式为 $4 \arctan \left(\frac{9\sqrt{2}}{274+85\sqrt{10}+28\sqrt{79}+10\sqrt{790}} \right)$ 。



此页故意留空



Problem H. 数字子序列

输入：标准输入
输出：标准输出
时间限制：8 秒
空间限制：512 MiB

给定一个长度为 n 的数字串 D ，其中每一位都是 $0 \sim 9$ 的数字。一个数字子序列是若干个互不相交且从左到右排列的非空连续子串

$$D[l_1..r_1], D[l_2..r_2], \dots, D[l_k..r_k], \quad 1 \leq l_1 \leq r_1 < l_2 \leq r_2 < \dots < l_k \leq r_k \leq n$$

每个被选择的子串按照通常的十进制表示解释为一个整数，数字子序列的长度定义为它包含的整数个数，即上式中的 k 。

如果一个数字子序列对应的整数序列 $x_1, x_2, \dots, x_k$ 满足 $x_1 < x_2 < \dots < x_k$ 则称它是上升的。请你求出给定数字串 D 的最长上升数字子序列的长度。

输入格式

第一行输入一个正整数 T ($1 \leq T \leq 500$)，表示数据组数。接下来按如下格式输入 T 组数据：

第一行输入一个数字串 D ，设 D 的长度为 n ，保证 $1 \leq n \leq 10^5$ 。

保证所有数据中 n 的总和不超过 5×10^5 。

输出格式

对于每组数据，输出一行一个整数，表示最长上升数字子序列的长度。

样例

标准输入
1 271828182845904523536028747135266249775724709369995

标准输出
17

提示

对于样例，一种最优方案是选择

2, 7, 8, 18, 28, 45, 52, 53, 60, 287, 471, 526, 624, 977, 5724, 7093, 69995

因此答案为 17。



此页故意留空



Problem I. 数组

输入：标准输入
输出：标准输出
时间限制：2 秒
空间限制：512 MiB

给定长为 n 的序列 $A = (a_1, a_2, \dots, a_n)$ 和长为 2^w 的数组 $H[0..2^w - 1]$ 。对于 $1 \leq i \leq n$ ，定义

$$C_i = \sum_{1 \leq b_1, b_2, \dots, b_m \leq n} H \left[\left(\sum_{j=1}^m a_{b_j} \right) \bmod 2^w \right] \cdot \left(\sum_{j=1}^m [b_j = i] \right)^k.$$

其中，外层求和遍历所有 $B = (b_1, b_2, \dots, b_m)$ ($1 \leq b_j \leq n$)， $[\cdot]$ 为艾弗森括号。

求 $C_1, C_2, \dots, C_n$ 对 998 244 353 取模的结果。

输入格式

第一行输入一个正整数 T ($1 \leq T \leq 35$)，表示数据组数。接下来按如下格式输入 T 组数据：

第一行输入四个整数 n, w, m, k ($1 \leq n \leq 5 \times 10^5, 0 \leq w \leq 19, 1 \leq m \leq 10^9, 1 \leq k \leq 20$)。

第二行输入 n 个整数表示 $A = (a_1, a_2, \dots, a_n)$ ($0 \leq a_i < 2^w$)。

第三行输入 2^w 个整数表示 $H[0..2^w - 1]$ ($0 \leq h_i < 2^w$)。

保证输入数据中 $\sum n, \sum 2^w$ 均不超过 10^6 。

输出格式

共 T 行，对于每组数据，输出一行 n 个整数，依次为 $C_1, C_2, \dots, C_n$ 。

样例

标准输入	标准输出
2 2 2 2 1 0 1 1 1 2 2 4 2 1 1 0 1 2 3 3 3 2 0	4 6 3 3 2 0

提示

本题输入输出量较大，建议使用较快速的输入输出方式（如关闭流同步的 `cin / cout`）。
请注意常数因子对程序效率的影响。



此页故意留空



Problem J. 游戏

输入：标准输入
输出：标准输出
时间限制：2 秒
空间限制：256 MiB

Alice 和 Bob 在玩取石子游戏，有 n 堆石子从左到右排成一排，初始时从左到右第 i 堆有 x_i 颗石子。

Alice 和 Bob 轮流操作：

- 轮到 Alice 操作时，Alice 从最左边的堆选至少一个石子，把选中的石子移到第二左的堆。
- 轮到 Bob 操作时，Bob 从最右边的堆选至少一个石子，把选中的石子移到第二右的堆。

若轮到某人无法操作时，当前操作的人输掉游戏。若 Alice 和 Bob 都使用最优策略，你需要判断 Alice 是否有先手必胜策略。

输入格式

第一行输入一个正整数 T ($1 \leq T \leq 10^5$)，表示数据组数。接下来按如下格式输入 T 组数据：

每组第一行输入一个整数 n ($1 \leq n \leq 10^6$)。

第二行输入 n 个正整数表示每堆石子数 $x_1, x_2, \dots, x_n$ ($1 \leq x_i \leq 10^9$)。

保证输入数据中 $\sum n \leq 3 \times 10^6$ 。

输出格式

共 T 行，Alice 有先手必胜策略输出 YES，否则输出 NO。

样例

标准输入	标准输出
5	NO
5	YES
4 5 4 5 9	YES
3	NO
4 4 1	YES
2	
10 9	
5	
1 2 1 1 2	
5	
2 1 1 2 1	



提示

本题输入输出量较大，建议使用较快速的输入输出方式（如关闭流同步的 `cin / cout`）。



Problem K. 森林

输入：标准输入
输出：标准输出
时间限制：1 秒
空间限制：512 MiB

设有 n 个点的完全图为 $K_n = (V, E)$ 。

给出 k 个 K_n 的子图 $G_1 = (V, E_1), G_2 = (V, E_2), \dots, G_k = (V, E_k)$ ，满足：

- $G_1, G_2, \dots, G_k$ 均无环。
- $E_1, E_2, \dots, E_k$ 的大小均为 m 。

记第 i 个子图 G_i 的第 j 条边为 $e_{i,j}$ 。

定义边集 S 是好的，当且仅当：

- 存在集合 $X_1, X_2, \dots, X_m$ ，满足 $X_i \subseteq \bigcup_{j=1}^k \{e_{j,i}\}$ 且 $1 \leq |X_i| \leq C_i$ ，使得 $S = \bigcup_{i=1}^m X_i$ 。
- 边集 S 导出的图 $G_S = (V, S)$ 无环。

求最大的集合 S 的大小。

输入格式

第一行输入一个正整数 T ($1 \leq T \leq 100$)，表示数据组数。接下来输入 T 组数据：

第一行输入三个整数 n, m, k ($1 \leq n, m, k \leq 100$)。

第二行输入 m 个整数表示 $C_1, C_2, \dots, C_m$ ($1 \leq C_i \leq k$)。

接下来 k 行，第 i 行输入 $2m$ 个整数 $u_{i,1}, v_{i,1}, \dots, u_{i,m}, v_{i,m}$ 表示子图 i 的所有 m 条边。

保证所有数据中， $\sum n, \sum m, \sum k \leq 200$ ，且 $1 \leq u_{i,j}, v_{i,j} \leq n$ 。

输出格式

共 T 行，对于每组数据，输出一个整数表示答案。

样例

标准输入	标准输出
2	1
2 1 1	2
1	
1 2	
5 1 3	
3	
4 1	
1 3	
1 3	



此页故意留空



Problem L. 向量

输入：标准输入
输出：标准输出
时间限制：2 秒
空间限制：512 MiB

上一世，小 L 线性代数期末考试不会做这道题导致挂科，身败名裂，多年心血付诸东流。
这一世，小 L 突破重重关卡，成功将手机带进考场，把这道题拍给了豆包。
你作为被囚禁于豆包中的众多灵魂之一，被分配到解答这个问题。

已知 A 是一个 n 行 n 列的 01 矩阵，且矩阵中值 1 的位置的个数恰好为 m 。

v 是一个 n 维列向量，且 v 的每个分量均为 $[-10^9, 10^9]$ 内的整数。设向量 $u = (I + kA)v$ ，其中 I 为 n 阶单位矩阵， k 是一个给定的常数。请你根据给定的 n, k, m, A, u ，求出列向量 v 。

若存在多个满足条件的 v ，请输出其中字典序最小的答案；若不存在满足条件的 v ，则输出 No Solution。

对于两个 n 维向量 v_1, v_2 ，称 v_1 的字典序小于 v_2 的字典序，当且仅当存在 $1 \leq i \leq n$ ，使得 $\forall 1 \leq j < i$ 都有 $v_{1,j} = v_{2,j}$ ，且 $v_{1,i} < v_{2,i}$ 。

输入格式

第一行输入一个正整数 T ($1 \leq T \leq 10^5$)，表示数据组数。

接下来按如下格式输入 T 组数据：

第一行输入三个整数 n, k, m ($1 \leq n, m \leq 10^6, 2 \leq k \leq 10^6, \sum n, \sum m \leq 2 \times 10^6$)。

第二行输入 n 个整数表示 $u^T = (u_1, u_2, \dots, u_n)$ ($|u_i| \leq 10^{18}$)。

第三行输入 m 个整数 $x_1, x_2, \dots, x_m$ ($1 \leq x_i \leq n$)。

第四行输入 m 个整数 $y_1, y_2, \dots, y_m$ ($1 \leq y_i \leq n$)。

第三行和第四行表示矩阵 A 中值为 1 的位置，即 $a_{x_i, y_i} = 1$ 。

保证每组数据内不存在重复的 (x_i, y_i) 对。

输出格式

共 T 行，对于每组数据，输出 n 个整数，表示 $v^T = (v_1, v_2, \dots, v_n)$ 。

若无解或不合法，输出一行一个字符串 No Solution。



样例

标准输入	标准输出
2	1 1
2 2 1	No Solution
3 1	
1	
2	
2 2 2	
1 0	
1 2	
2 1	

提示

本题输入输出量较大，建议使用较快速的输入输出方式（如关闭流同步的 `cin / cout`）。



Problem M. 彩色匹配

输入：标准输入
输出：标准输出
时间限制：15 秒
空间限制：512 MiB

给定一个二分图 $G = (L, R, E)$ ，其左右两部各有 n 个点，之间有 m 条边，每条边被染成蓝色或红色。

一个完美匹配是 n 条两两没有公共端点的边，使得每个左部点和每个右部点都恰好被匹配一次。

请你找出一组完美匹配，使得其中恰好有 k 条红边。

输入格式

本题有多组测试数据。

第一行输入一个整数 T ($1 \leq T \leq 60$)，表示测试数据组数。

接下来按如下格式输入 T 组数据：

第一行输入三个整数 n, m, k ($1 \leq n \leq 120, 0 \leq m \leq n^2, 0 \leq k \leq n$)，表示二分图左右两部各自的点数、图的总边数，以及完美匹配中要求的红边数。

随后 m 行，每行输入三个整数 u, v, c ($1 \leq u, v \leq n, c \in \{0, 1\}$)，表示左部点 L_u 与右部点 R_v 之间有一条边。

- $c = 0$ 表示蓝边；
- $c = 1$ 表示红边。

保证同一组数据中不会出现 u, v 均相同的两条边，至多 10 组数据 $n > 50$ 。

输出格式

本题使用 Special Judge 进行评测。

对于每组数据，输出一行。如果不存在红边数恰好为 k 的完美匹配，输出 **-1**。

否则输出 n 个用空格分隔的整数 $p_1, p_2, \dots, p_n$ 。其中 p_i 表示左部点 L_i 与右部点 R_{p_i} 匹配。

输出必须满足：

- p_i 均为 1 到 n 之间的整数，且两两不同；
- 对所有 i ，边 (L_i, R_{p_i}) 存在；
- 恰好有 k 条被选中的边为红边。



样例

标准输入	标准输出
3	1 2
2 4 1	1 2
1 1 1	-1
1 2 0	
2 1 0	
2 2 0	
2 2 2	
1 1 1	
2 2 1	
2 2 1	
1 1 0	
2 2 0	